



## Sincronizando los datos

### Transcripción

[00:00] Hola todos y todas. Bienvenidos y bienvenidas una vez más a un nuevo video de Git. Ahora lo que tenemos es esta estructura de acá, tenemos el usuario Bruno, tenemos al usuario Ana y tenemos un servidor, y lo que tenemos que hacer es hacer esto que dice acá esta flecha, hacer que Bruno envíe datos al servidor.

[00:20] Para eso vamos a ir a nuestra terminal, estamos dentro de la carpeta de Ana. Entonces vamos a hacer un `cd ../.` y acá vamos a ir a ciudad Bruno. Estamos dentro de nuestro proyecto. Vamos a hacer un `clear` para limpiar un poco la pantalla.

[00:41] Entonces voy a hacer un `git remote` acá, damos `enter` y vemos que tenemos nuestro servidor local. Entonces lo que yo tengo que hacer es agarrar los datos que tengo dentro de mi repositorio que tenemos acá adentro de Bruno y empujarlos hacia el servidor. Y dije empujar porque en Git nosotros usamos el comando `push`, que es empujar en inglés.

[01:08] Entonces lo que tenemos que hacer es usar `git push` pero ahora la pregunta es desde dónde y hacia dónde vamos a enviar estos datos. Nosotros vamos a enviar los datos a servidor local y, ¿cuál branch? Nuestra única branch que tenemos por ahora es `master`, ya más adelante vamos a hablar un poco más sobre branches.

[01:34] Damos `enter`, y vean que tenemos todo esto es que está todo bien, se envió el 100%, está todo bien y se creó una nueva rama llamada `master` dentro

de nuestro servidor que es acá, esa ruta dentro del servidor. Entonces, hemos enviado los datos a nuestro servidor y ahora vamos a loguearnos como Ana.

[02:00] Vamos a hacer un `cd.. enter`. Acá vamos a hacer `cd Ana/proyecto`, okay. Damos `ls` y acá debería tener las cosas que envió Bruno al servidor pero no lo estamos teniendo. ¿Por qué? Porque lo que hemos hecho es que Bruno envíe los datos al servidor, pero Ana ahora tiene que traer esos datos del servidor a su repositorio de Ana. Entonces vamos a hacer eso.

[02:32] Vamor a ver cuáles son los repositorios que tiene Ana. Vamos a hacer `git remote`, damos `enter` y vemos que tenemos acá un `origin`. ¿Qué es ese `origin`? Ese `origin` es el nombre que pone Git por defecto cuando nosotros agregamos un repositorio. Entonces lo que vamos a hacer ahora es cambiar ese nombre `origin` para `servidor local` como tiene Bruno para mantener las apariencias.

[02:57] Entonces lo que vamos a hacer es utilizar el comando `git remote rename origin local`, damos `enter`. Parece como que no pasó nada, pero si hacemos `git remote`, damos `enter` y ahora aparece `local`. Acá cometí un error, lo llamé de `local`. Vamos a hacer de nuevo utilizando la flechita para arriba, vamos a colocar `git remote local`, y vamos a llamarlo de `servidor local`, como tiene Bruno.

[03:34] Ahora sí, si hacemos `git remote` aparece `servidor local`. Volvemos al `git pull`, vamos a agregar el repositorio `servidor local`. Entonces vamos a hacer un `git pull servidorlocal`, ahí. Y dentro de ese `servidor local`, ¿cuál rama tenemos que traer? En este caso vamos a traer la rama `master` que está dentro de ese `servidor local`. Es la única que estamos trabajando.

[04:05] Ya como les dije vamos a hablar más adelante un poco más sobre ramas. Ahora vamos a hacer `master`, damos `enter` y vemos que creó una nueva rama dentro de nuestro proyecto de Ana llamada `master`, y ahora si nosotros damos un `ls`, vemos que tenemos nuestro `index`.

[04:26] Vamos a ir un poco a Visual Studio Code, acá les voy a mostrar la carpeta Bruno y tenemos la carpeta Ana. Vean que el archivo de Bruno está igual que el archivo de Ana, tienen los mismos datos. Perfecto. Un detalle más a anotar es que dentro de la carpeta de Ana, del proyecto de Ana, nosotros no tenemos este a.config ni la carpeta. Recuerdan que hemos colocado dentro del git ignore esos datos.

[05:01] Entonces Ana no va a sincronizar ese a.config ni esa carpeta. Ahora vamos a hacer por ejemplo que Ana, dentro de Ana proyecto, ella modifica ahora por ejemplo el título de Integración Continua. Digamos que lo coloca más completo y coloca el nombre de Integración Continua: Madurez y Productividad en el Desarrollo de Software.

[05:42] Supongamos que Ana agregó datos a este archivo. Vamos a guardar con un "Ctrl + S" y ahora volvemos a nuestra terminal. Estamos dentro del proyecto de Ana. Vamos a hacer un git status para ver si está todo bien y vean: acá aparece que se modificó el archivo index.html. Entonces lo que vamos a hacer ahora es enviar esas modificaciones al servidor de nuevo.

[06:12] ¿Cómo hacemos eso? Bueno, vamos a primero agregar con git add. Esperen que primero limpio la pantalla para que quede un poco más limpio. Vamos a hacer git add index.html. Damos enter. Vamos a hacer un git commit -m. Todo esto ya lo hemos visto. Vamos a poner: "Renombrando curso de Integración Continua".

[06:46] Vamos a dar enter y ya modifiqué. Y ahora que lo agregué, ¿será que ya Bruno consigue ver esas modificaciones? Bueno, en realidad no, porque todavía no envié al servidor los datos, ese nuevo commit. Entonces lo que vamos a hacer es hacer un git push servidorlocal master. Vemos que ahí envió los datos y ahora lo que vamos a hacer es volver para la carpeta Bruno.

[07:19] Entonces vamos a hacer un cd, espacio, ../. enter, ls, y acá tengo todas mis carpetas, vamos a cd Bruno, enter y ya estamos en Bruno. Una vez que

estoy en Bruno vamos a traer los datos del servidor porque Ana nos dijo:  
"Bruno, modifiqué el archivo index, entonces podés hacer ya un pull".  
Entonces lo que vamos a hacer es un git pull servidorlocal master.

[07:57] Vemos que nos trajo datos, hizo un pull del master dentro del servidor, nos dice que ha cambiado un archivo, se agregó una línea y se quitó una línea. Si nosotros hacemos un git log, espacio, -pero, enter. Vemos cual línea se modificó, se quitó esa línea de acá y se agregó esa "Integración Continua: Madurez, etcétera".

[08:27] Vamos a salir de ese git log -p poniendo :q, enter. Ya hemos salido y con esto ya vemos que ya podemos sincronizar los datos y modificaciones de los códigos que nosotros estamos creando entre los integrantes de nuestro equipo. Pero, ¿y si no quiero crear un servidor local, mucho menos compartir una carpeta en la computadora?

[08:50] Si quiero poner los datos en un servidor en Internet. ¿Será que hay servicios como repositorios Git online? Bueno, vamos a hablar un poco de eso en el próximo video.